

Description

This Mini-Project is Part of Rakamin Academy's Data Science Bootcamp.

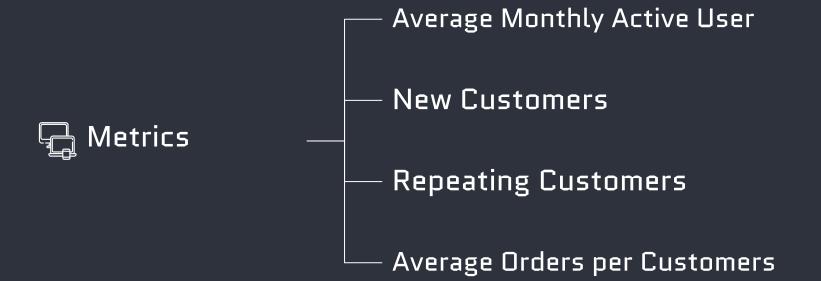
Viewing and analysing an ecommerce business, whether it is growing, stagnating or even declining.

One of the metrics used to measure the performance of an eCommerce business is the customer activity as they interact with the eCommerce platform.

In Assignment 2

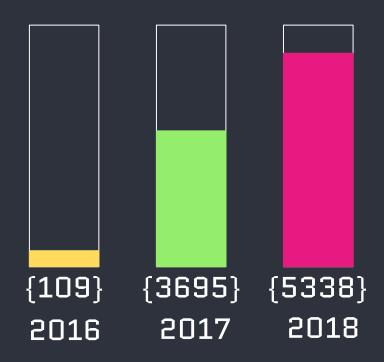
I analysed several metrics related to customer activity, such as the number of active customers, the number of new customers, the number of repeat customers

and also the average number of transactions per customer per year.



Average Monthly Active Users

	year double precision	average_monthly_active_user numeric
1	2016	109
2	2017	3695
3	2018	5338



Average Monthly Active Users

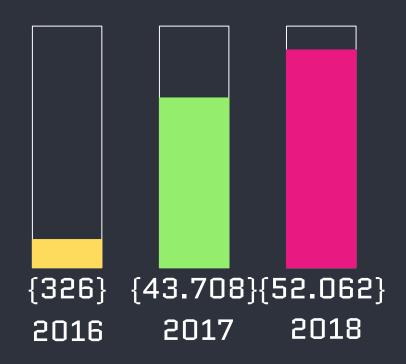
The first step is to create a temporary table with a subquery in the from clause. The table is used to count the number of unique customers who actively place orders each month, called Monthly Active Users (MAU).

This is called the Monthly Active User (MAU). The date_part function is used to extract the extract the timestamp components, in this case the year and month components are extracted as needed. are extracted as required.

After obtaining the MAU number for each month, a larger aggregation can be performed to obtain the average MAU number. A larger aggregation can be performed to obtain the average MAU for each year. average calculation, grouped by year.

New Customers

	year double precision	new_customers bigint
1	2016	326
2	2017	43708
3	2018	52062



New Customers

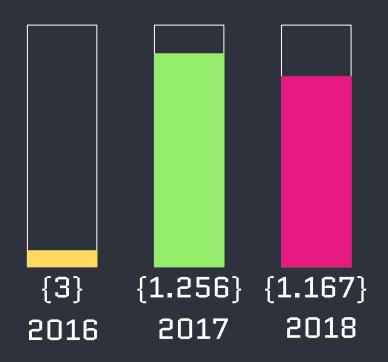
The temporary table contains information about the first purchase time stamp for each customer. for each customer. This can be easily obtained by using the MIN aggregation function on the order_purchase_timestamp column to get the earliest date for each customer.

Having obtained the first order date for each customer, the next step can be done by extracting the year and then calculating if there is a purchase date for each customer.

The next step can be done by extracting the year and then calculating how many customers there are for each year. for each of these years. This last figure shows how many new customers there are in each year. in each year.

Repeating Customers

	year double precision	repeating_customers bigint
1	2016	3
2	2017	1256
3	2018	1167



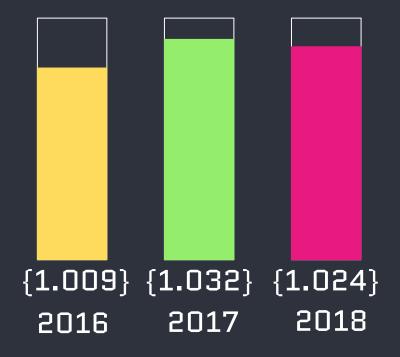
Repeating Customer

In this Subtask 3, a temporary table was created showing the number of orders placed by each customer per year. Each customer per year, grouping COUNT calculations based on year and customer_unique_id.

customer_unique_id. The aim is to get customers who place repeat orders, using the HAVING filter to get customers who place more than 1 order. Aggregation of COUNTs grouped by year can then be performed to calculate how many customers make repeat orders (order more than 1) for each year.

Average Orders per Customers

	year double precision	avg_orders_per_customers numeric	
1	2016	1.009	
2	2017	1.032	
3	2018	1.024	



Average Orders per Customers

After a table containing order frequency information for each customer each year, further aggregation can be done by running AVG on the Frequency column.

Further aggregation can be done by performing AVG on the Frequency column, which is then grouped by year. grouped by year. This will give the average number of orders for each customer in each year.

	year double precision 6	average_mau numeric	new_customers bigint	repeating_customers bigint	avg_orders_per_customers numeric
1	2016	108.67	326	3	1.009
2	2017	3694.83	43708	1256	1.032
3	2018	5338.20	52062	1167	1.024